Understanding Privacy-Related Questions on Stack Overflow

Understanding Privacy-Related Questions on Stack Overflow

Mohammad Tahaei, Kami Vaniea, Naomi Saphra
School of Informatics
University of Edinburgh
{mohammad.tahaei, kami.vaniea, naomi.saphra}@ed.ac.uk

ABSTRACT

We analyse Stack Overflow (SO) to understand challenges and confusions developers face while dealing with privacy-related topics. We apply topic modelling techniques to 1,733 privacy-related questions to identify topics and then qualitatively analyse a random sample of 315 privacy-related questions. Identified topics include privacy policies, privacy concerns, access control, and version changes. Results show that developers do ask SO for support on privacy-related issues. We also find that platforms such as Apple and Google are defining privacy requirements for developers by specifying what "sensitive" information is and what types of information developers need to communicate to users (e.g. privacy policies). We also examine the accepted answers in our sample and find that 28% of them link to official documentation and more than half are answered by SO users without references to any external resources.

Author Keywords

Software Developers; Usable Privacy; Stack Overflow.

CCS Concepts

•**Security and privacy** → **Human and societal aspects of security and privacy; Usability in security and privacy;** •**Software and its engineering** → *Software creation and management;* •**Social and professional topics** → **Privacy policies;**

INTRODUCTION

When designing software, developers have to make a range of decisions that impact many aspects of the software such as efficiency, maintainability, and privacy. Developers located in large organisations may have access to dedicated staff with training in such topics to assist them, but for many developers, they are expected to incorporate these features into their code on their own. This observation begs the question of how developers manage privacy in software as well as how they interpret and think about privacy-related coding issues.

CHI '20, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6708-0/20/04.
http://dx.doi.org/10.1145/3313831.3376768

Security and privacy can be challenging for developers to get right, even with the support of tools [12, 22, 49]. Developer errors are a common source of vulnerabilities [24] with many causes ranging from APIs with poor developer support [1, 47] to static analysis tools that produce too many false positives [38]. Privacy can also be challenging for small organisations where their income depends privacy-unfriendly monetisation methods such as ad networks [14, 46].

Privacy, as a social norm, can define how security is being implemented as a technological requirement [15, 23]. While prior research has found several reasons for developers' poor security practices [30, 53], we know comparatively little about the privacy challenges and concerns they face. Efforts to introduce privacy into technical levels such as privacy by design [35] are still nascent, and there is a gap between these frameworks and how software developers approach privacy [33].

Stack Overflow [69] (SO) is one of the largest developer Q&A platforms and defines itself as "an open community for anyone that codes." It attracts a wide range of developers who ask questions about programming, security, and data management [18, 56, 76]. SO's dataset has been heavily used for research on such topics as: what factors makes it a successful Q&A platform [45], security issues developers face and how they interact and build knowledge around it [43, 76], and the negative impact of SO code snippets in software security [2].

Our research combines techniques from the literature on SO analysis with questions about the privacy-related tasks of developers. Our research questions are: 1) What topics do SO users associate with the word "privacy"? 2) What or who is pushing SO users to engage with privacy related topics?

To answer our research questions, we collect SO questions that mention "privacy" in the title or tags and then apply topic modelling and manual qualitative analysis methods. We find that developers ask questions when dealing with permissions, access control, encryption, and privacy policies. Similar to other works [18, 56, 71], we look at question types such as "how" questions that ask for instructions and help, "conceptual" when they look for advice and suggestion in early stages of development, "errors" which includes crashes, and "unexpected" which includes surprises from updates or features being added or removed. We further analysed the accepted answers, which shows that 28% of those link to official documentation.

RELATED WORK

Stack Overflow

SO is aimed at software developers, covering various topics such as website development, databases, version control, and security [16]. It has an Alexa rank of 43 [4] and more than 50 million unique visitors per month (as of September 2019) [66].

SO Users: SO surveys developers every year and publishes the results. The 2019 survey includes responses from 88,883 software developers from 179 countries in which 85.6% of respondents are SO users. Most respondents "said they are professional developers or who code sometimes as part of their work, or are students preparing for such a career" [67]. Over 85% visit SO at least a few times per week, with over 60% visiting every day and 96.9% using it to find answers to specific questions. 73.9% are employed full-time at companies whose size ranges from "just me" to "10,000 or more employees".

Impact of SO on software security: Developers utilise SO knowledge and code snippets to build their projects [6, 54, 74]. A study of 289 open-sources projects showed that 30.5% of projects contained code matching code found on SO with some modification [74]. However, code reuse from SO can also introduce vulnerabilities [2, 3, 26]. For example, Fischer et al. found that snippets from SO questions that contained security-related code were observed in 15.4% of applications on Google Play (1.3m apps), and 97.9% of those apps had at least one snippet with insecure code [26].

Researchers have also studied the topics developers talk about; including analysis with natural language processing techniques (NLP) [5, 16, 18, 56, 71, 76] and manual qualitative techniques [18, 36, 43, 44, 47, 48, 52, 55, 71]. For example, an analysis of questions about Puppet, a configuration language tool, shows a need to support Puppet syntax error finding [55].

Topics: Prior topic modelling of security SO questions found five main categories: web security (51%), system security (19%), cryptography (17%), software security (9%), and mobile security (4%); with popular subjects including: password, hash, signature and SQL injection (out of 30,054 posts) [76]. Such outcomes can help both industry and researchers to understand better the challenges developers are facing. For example, injection (such as SQL, NoSQL, LDAP) and broken authentication such as passwords, keys, and session tokens are the two top risks in OWASP's ten most critical web application security risks [50], which are similar to the findings of Yang et al. who also studied SO questions [76].

Question types: Questions posed on SO can be a good indicator of the areas of development SO users require guidance on. For example, they ask questions around library features, then clarify optimal implementations once they are confident with basic functionality. They will ask for solutions, workarounds and explanations when their code has errors or unintended features. Finally, they may ask for improved solutions with best practices [5, 18, 48, 56, 71].

Privacy and developers

There is no unified cross-discipline definition of privacy [61]. Daniel J. Solove describes privacy as "too complicated a concept to be boiled down to a single essence" [62, p.485], so he instead made a taxonomy of activities that potentially can be harmful to privacy: *information collection* (e.g. surveillance), *information processing* (e.g. identification), *information dissemination* (e.g. disclosure), and *invasion* (e.g decisional interference) [62]. In the engineering realm, privacy is defined as a set of requirements collected from stakeholders. For instance, software developers are expected to pay attention to activities that can threaten privacy in information systems such as data transfer, storage, and processing [64]. *Notice and choice*, *privacy-by-policy*, *privacy-by-architecture* [64], and *Privacy by Design* (PbD) [20, 21, 35, 39, 73] are some examples among many other frameworks which include practices and guidelines to bring privacy into the design space.

Prior research uses PbD to understand the privacy practices of software developers and development [17, 31, 33, 35, 59]. Ann Cavoukian, who coined the term, describes PbD as "assures an end-to-end chain of custody and responsibility right from the very start" [21, p. 406]. PbD thus aims to bring privacy into the system development process [32].

PbD for developers: Semi-structured interviews with 27 developers showed that they interpreted the concept of privacy as a set of smaller concepts, such as security, confidentiality, purpose specification, and consent. In contrast, concepts such as notice, minimisation, and rectification were not mentioned by many participants. Participants reported that they were familiar with other privacy concepts, such as user transparency and automatic expiration date, yet admitted they used these technologies infrequently [33].

Interviews with senior engineers show that privacy is seen as a burden which no one views as their own responsibility as well as a concept that is hard to define because it is wrapped up in legal jargon [17]. These results are similar to a study that was carried out 15 years previous, which indicates stagnation in engineer's mindsets [17]. Beyond interviews, a discourse analysis of two mobile developer forums for privacy relevant conversations found that developers of these forums were concerned about how third-parties are collecting data, the privacy implications of features requested by end-users, and the legal consequences of their actions [60].

Developers are one of several privacy decision-makers: The costs and effects of developers' choices and mistakes in software systems can be enormous [25, 27]. These decisions, however, are influenced by the choices made in designing the systems they are dependent on, including platforms, APIs, and human organisations. For example, mobile platforms shape the privacy mindsets of their developers; iOS developers are more concerned about "notice and consent" as Apple promotes it, while Android developers advertise privacy as an extra feature to stand out in the market [31].

API design influence developer choices. A lab study with developers given the choice between coarse and precise location APIs found that they chose the coarse location option [37], providing more privacy. Nudges and help from documentation [13], models [42], and IDE plugins [41] can also assist developers in privacy-friendly software development.

Organisation internals are another key factor in the security and privacy practices of developers [10, 11, 14, 33, 75]. For example, the size of the company influences the privacy behaviour of developers; larger companies are more concerned about having a privacy policy (PP) [14]. Moreover, some developers follow practices suggested by their employers, such as programming languages and tools [11]. They also benefit from the advice of security experts in their organisation [11].

Latent Dirichlet Allocation (LDA)

LDA [19] is a common method of topic modelling. It is an unsupervised method, meaning that the topics are not labelled by humans, but are discovered naturally through patterns of clustering in the data. For example, LDA might discover that documents fall into two topics, one in which typical words include (baseball, bat, pitcher), and another in which these common words are (neural, Gaussian, marginalised). A human annotator is needed to label these topics as "baseball" and "machine learning", as the model does not assign labels. Note that the word "statistics" could easily signify either topic; vocabulary is not exclusive to a single topic, but has different distributions according to topic. LDA models text generation as a two-step process: first, a mixture of topics is sampled through the Dirichlet distribution, then a mixture of vocabulary items is sampled from the Dirichlet distribution associated with each topic. The model assumes that the words in a document are sampled by selecting a topic from the mixture of topics and a word from the mixture of words associated with that topic. We interpret these topics by inspecting the words most indicative of each topic. We take advantage of this automation to analyse a larger dataset than is feasible with human annotation. The approaches in Section 2.1 use LDA to find topics in SO questions [5, 16, 18, 56, 71, 76].

Our contribution

A systematic literature review of developer-centred security shows that few papers study the intersection of developers and privacy, and further research is needed in this area [70]. Our work contributes to this research area by studying SO privacy-related questions using both automatic (LDA), and manual (qualitative coding) approaches. Our approach is a bottom-up analysis which builds upon questions developers asked when they faced a privacy-related problem or felt the need to dispel confusion on a related topic. This study complements existing interview work in the privacy space.

METHOD

SO data collection

We collected three data sets from SO; each composed of question and answer text as well as metadata such as the number of views and votes. *SO-all* is the set of all SO questions. We use this set to provide comparison statistics. *SO-privacy* is the set of all SO questions where the word "privacy" appeared in either the question title or tags (n=1733). The term "privacy" was selected after iterating on several alternatives and finding minimal improvement of quality. We use this set for most of the quantitative analysis, including the LDA topic model. Finally, *SO-privacy-rand* is a set of 315 questions randomly selected from *SO-privacy* and is used in the manual qualitative

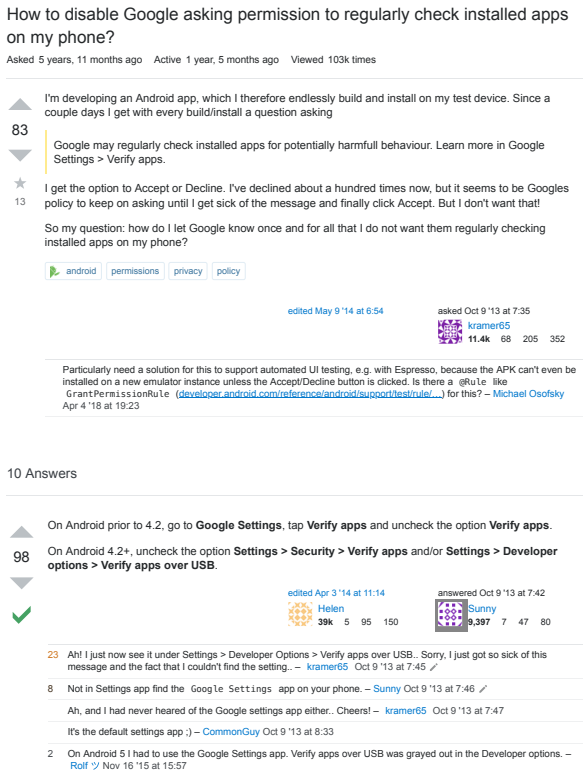

Figure 1. A sample privacy-related question with an accepted answer.

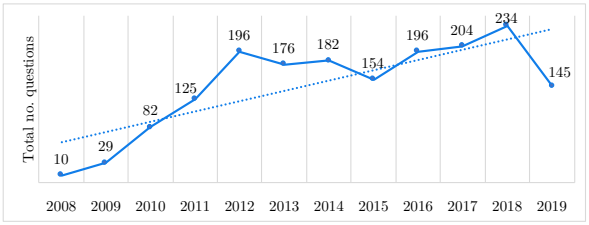

Figure 2. Count of questions mentioning privacy per year (SO-privacy).

coding. Figure 1 shows a sample privacy-related question. All data was collected using the Stack Exchange Data Explorer [65]. The research was conducted in accordance with our institute's ethics procedures.

Looking at SO-privacy, the first question was created on 02 Aug. 2008 (for SO-all it was on 31 July 2008), and the most recent was created 17 Aug. 2019. 1,428 questions have at least one answer, and 790 have an accepted answer. Tables 1 and 2 provide a comparison between the data sets in terms of users and questions. Figure 2 shows the number of questions asked by year and Figure 3 shows the top 50 tags assigned by askers in SO-privacy.

Topic extraction - LDA

Documents were formed from SO-privacy by concatenating the question title and body, lemmatised with stop words removed using `spaCy` [63]. All code samples and URL details were removed so the topics would be based only on natural language data. We ran a bigram LDA at 2000 iterations, with a

	Reputation[1]	*Views*[2]	*Up Votes*[3]	*Down Votes*[3]
All users (10,901,490)				
Avg	106	14	11	1
SD	2,312	708	180	361
SO-all question askers (3,501,541)				
Avg	2,631	389	270	29
SD	12,929	3,780	903	428
SO-privacy question askers (1,684)				
Avg	3,430	448	268	49
SD	18,453	1,974	706	648
SO-privacy-rand question askers (312)				
Avg	4,889	602	312	110
SD	25,413	2,927	785	1,135

Table 1. Stats for SO users and users in our subsets. [1]Can be gained by posting good questions and answers. [2]Number of times the profile is viewed. [3]How many up/down votes the user has cast.

	Score[1]	*Views*	*Answers*	*Comments*	*Favourites*[2]
SO-all (18,123,431)					
Avg	2	2,279	2	2	3
SD	23	18,419	1	3	20
SO-privacy (1,733) - Used for LDA and qualitative analysis section					
Avg	3	1,416	1	2	3
SD	16	7,338	2	2	11
SO-privacy-rand (315) - Used for coding findings section					
Avg	4	1,378	1	2	3
SD	25	5,281	1	2	8

Table 2. Stats for questions. [1]The difference between up votes and down votes. [2]Similar to bookmarking a question.

variety of topic counts, from 5 to 60. After discussions among researchers, we selected 15 topics as the best setting.

Qualitative analysis

Two researchers first independently read through 40 questions drawn at random from SO-privacy, and also reviewed the output of the LDA topics. Then during multiple discussion sessions and meetings they shared their observations and identified four interesting elements of the questions deserving of further analysis: 1) the question type, based on existing taxonomies [5, 16, 18, 71]; 2) the driver that makes the user need to ask the question (e.g. compiler error, client requirement, or Facebook warning); 3) the aspect of privacy that the question relates to (e.g. setting app permissions); 4) accepted answers.

Question type

In prior work, question type focuses on the shape of the question, such as "how do I...?" questions. After reviewing both the question types found in prior work [5, 16, 18, 71] and the shapes of questions found in the SO-privacy set, we narrowed the question types to: 1) conceptual questions that ask for higher level explanation, as well as moral, legal, and regulatory advice; 2) unexpected behaviour the asker wants to be explained; 3) error questions where the asker provides an error and asks how to fix it or why it is happening; and 4) questions looking for instructions, solutions, and best practice.

Coding procedure: After the question type codebook was solidified, both researchers coded 10% of the data. The question types inter-rater reliability kappa was 70%. One researcher coded the rest of the data for question types, and the other researcher coded another 10% to make sure they did not drift apart and have a similar understanding of the data. Their final kappa was 77% which is considered as a good agreement [57].

Drivers

A driver is the event, technology, or motivation that caused the asker to post a question on SO. Some drivers are expected, such as getting a compiler error, while others are more unique to our data, such as concern over how to comply with the General Data Protection Regulation (GDPR). Our practice questions cited many reasons for interacting with privacy, such as requirements from clients, concern about laws, and the development platform (e.g. Facebook) giving privacy warnings that prevented code deployment. Unlike question type, drivers were quite varied and not easy to classify a-priori. Therefore, we decided to use open coding. One researcher went through all the questions and provided one or more open codes. A second researcher did the same for 10% of the dataset. We do not report the kappa values as they were open coded.

The two researchers then completed a thematic analysis [40] of the driver codes, resulting in four themes: 1) feedback from platforms such as operating systems (OS) or companies (e.g. Facebook, Google Play, Apple Store), 2) personal concerns and business reasons (e.g. company or client requirements), 3) laws and regulations such as GDPR, and 4) too vague or unclear to code properly.

Privacy aspect

The privacy aspect of a question describes how it relates to the concept of privacy. SO questions can be complex and contain multiple parts, not all of which involve privacy. For example, an asker wants to make sure users scroll to the bottom of the PP page before the "accept" button activates, but is having trouble with the way the fonts are showing on the page. In this case, the privacy aspect is ensuring users read the PP. Similar to the drivers, privacy aspects appeared to have a wide range which was hard to categorise a-priori. Therefore, we decided to open code the privacy aspect. Because aspects seemed to involve both a subject (PP, camera) as well as an action (create, change, use), coders were encouraged to create open-codes that contained both subjects and actions, where appropriate. For example, "create a PP" or "read camera permission state". As with the drivers, one researcher open coded the whole dataset, and the other coder did the same for 10%. Two researchers then grouped the codes into themes using thematic analysis [40]. We do not report the kappa values as they were open coded.

Accepted answers

One researcher analysed the accepted answers, that is "When a user receives a good answer to their question, that user has the option to "accept" an answer." [68], and coded them into these categories: 1) provides a solution, explanation, advice, opinion, sample code by an SO user, 2) links to another SO question, 3) when there is a link to an official documentation with or without any further explanation, and 4) links to an unofficial resource with or without any further explanation.

Figure 3. Top 50 most commonly used tags by users (SO-privacy).

Topic label	Top five words
1. Access to and read contents	app, application, use, android, privacy
2. Set the privacy field	user, privacy, like, page, facebook
3. App purchase and user registration	device, ios, cloud, feature, access
4. Privacy and permission settings and dialogues	app, user, privacy, access, ios
5. Crash reporting, analytics tools, and trackers	crashlytics, tracker, integrate, news, advertiser
6. PPs in Google Play and Android	app, policy, privacy policy, privacy, store
7. Concerns about using Google services	google, button, use google, ad, click
8. Publicity of sensitive data in code repositories	analytics, firebase, repository, google analytics, git
9. Design a db schema with privacy settings	table, privacy, column, transaction, mysql
10. Privacy values in Facebook, YouTube, and plists	privacy, post, set, facebook, api
11. Image privacy statements in Instagram and Windows	image, windows, statement, instagram, privacy statement
12. Store users' sensitive data securely	datum, user, use, address, information
13. Access to, create, and upload photos and albums	photo, album, picture, save, access photo
14. Private and public variables	file, private, privacy, use, code
15. Browsers errors (cookies and security settings)	use, privacy, website, browser, site

Table 3. LDA topics and the top 5 words in the topic (SO-privacy).

LDA FINDINGS

Table 3 shows the 15 LDA topic clusters generated from SO-privacy and their researcher-generated labels. The topics include a wide set of common security and privacy concepts such as access control, secure storage, data management, confidentiality, user consent, human factors, and tracking.

Apps are a large issue for developers, with terms like "app" occurring in multiple topics as well as the names of platforms that host apps such as Google and Facebook. App-related concepts such as permission settings are also a clear cross-cutting topic ranging from photo to location permissions.

Server-side issues also appear across several themes such as database design, handling of sensitive data, encryption, blockchain, handling account access, and storing passwords. The topics suggest that developers are encountering privacy not just as part of user-facing elements such as dialogues and alerts, but also in the design of their back-end infrastructure.

We also see a topic on public/private variable scopes (topic 14). Examination of questions associated with this topic show typographical errors where the user wrote "privacy" when they meant "private". While this topic is outside our scope, it is nice to see it neatly forms a distinct topic.

We find that "want" and "need", indicating that the asker is attempting a specific task as in "I need to access a file", are highly-ranked in topics 1, 2, 6, 11, 12, and 14. This behaviour can be connected to the qualitative question type *How*. "Thanks", a marker of politeness and possibly of discomfort with the SO community, is in the top 20 words indicating topics 1 (content access) and 10 (Facebook/YouTube/plists). This politeness divide may indicate differences in the background and persona of users interested in those topics.

CODING FINDINGS

Of the 315 randomly selected in SO-privacy-rand, 21 were excluded due to either being about private variables (scoping) or being too vague to understand. This section focuses exclusively on the remaining 294 questions. Because the research is bottom-up, we decided to use SO users' definition of "privacy" to understand their usage of the word rather than force our understanding of it. Consequently, the only posts we excluded were clear misspellings, most commonly those confusing "privacy" with the scoping word "private" as in public/private classes. This confusion was common enough to appear in the LDA results (topic 14 in Table 3). One interesting result of this user-lead definition is that some clusters are technically more security-focused or more UI-focused. But in all cases, the asker explicitly used the term "privacy" in the title or tags indicating that they thought the question was privacy related in some way.

Question types

How (186, 63%). These questions include instructions, solutions, best practices, and possibilities: *"I've used my personal email address for [Git repository] commits and I'm trying to set it to another one, before I make the repository public. [...] Is there a way to remove it from there, too, without losing my history?"* [13323759 - 2012].

Abstract or Conceptual (50, 17%). These questions ask for explanations, legal/policy/requirements advice, background information on a component or process, or further conceptual understanding. The asker's goal was to get advice about legal, policy, regulation, moral, or ethical implications: *"What is the hidden cost using these CDN services? If the script is not cached by the browser and it loads the script from google what could google potentially do with the information? Could it be usefully extrapolated in conjunction with other services such as search, analytics or adsense? Nothing is free, what's the catch?"* [10133816 - 2012].

Error (46, 16%). These questions quote a specific *error message* to understand the provenance of errors, exceptions, crashes, or even compiler errors. Includes warnings that are blocking progress to working project state (compilation, upload to store, etc.), including emailed "fix this" warnings from platforms. Questions containing compiler or similar errors are regularly observed on SO [18, 48, 56, 71]. Notably, the privacy questions quote warning messages from platforms: *"I still get privacy error with "NET::ERR_CERT_AUTHORITY_INVALID" in the browser when I hit the ELB url using https" Answer:"The issue is that you are using a self-signed certificate..."* [45295709 - 2017].

Unexpected (12, 7%). The asker wants some observed unexpected behaviour explained. Includes surprise due to features having been added or removed with a new version as well as unexpected behaviours that arise from OS or device revisions. A common example was the sudden addition or removal of permission dialogues when the developer switches to a new API version or different behaviour on different OS versions: *"I set microphone permission in info.plist file so record audion permission alert displaying in iOS 10.3.2 but its not appearing in iOS 10.3.3 devices."* [46297966 - 2019].

Drivers

The largest driver was *personal concerns, client or company requirements* (144, 49%). This finding is unsurprising, as this group includes cases where no driver is explicitly cited. Common external drivers in this group included a client requesting a feature, or commentary on what an app's end-users wanted. The second most common driver was feedback from a *platform* (136, 46%). This finding also makes sense since many third-party platforms, such as Facebook, have requirements that developers must follow. A common issue was that Google requires a URL to a PP if sensitive permissions are being used, resulting in several askers turning to SO to understand either why Google thought they were using sensitive permissions, or how to create a PP that met Google's requirements.

Drivers coming from *laws and regulations* (5, 2%) were least common. These included concerns around topics like GDPR or speculation about if an action was or was not legal. In SO-privacy-rand, we only observed question about EU regulations; however, in the broader SO-privacy sample, we observed mentions of regulations from other countries, such as the USA's Health Insurance Portability and Accountability Act (HIPAA).

Accepted answers

Answers contain sample codes, explanations, links to and quotes from other resources, opinions, hints, and screenshots. Out of 130 questions with an accepted answer: (76, 58%) were answered by *SO users*; (36, 28%) had a link to *official documentations*, websites, blogs; (17, 13%) had a link to *unofficial resources* such as websites, blogs, Wikipedia, an app, or a GitHub project; (4, 3%) were pointed to *another SO question*. Dual coding occurred in links to another SO questions in which two had a link to an official doc (included in the official group as well), one had a link to an unofficial doc (included in the unofficial category too), and one provided a link to another SO question.

For links to unofficial sources, Wikipedia and GitHub were most common. GitHub occurred eight times as a source for referring to issues and bugs, projects and pages that could be helpful to the asker. Wikipedia was used as a source for further details and explanation of concepts in five answers (with the concepts: AOL search data leak, flag fields, segmentation fault, ePrivacy Regulation (European Union), and P3P).

PRIVACY ASPECT THEMATIC ANALYSIS FINDINGS

Each subsection describes the (sub)themes, number of questions, percentage, and the number of question views associated with the theme. Table 4 gives an overview of the themes.

Access control (119, 40%, views: 103,654)

SO users often struggle to find information about updating and changing the privacy status of posts, images, and videos on social networks. They also ask about how to implement systems that have different levels of access control.

Dealing with privacy settings (59, 20%)

When SO users want to set the privacy field of a post, image, or video on social networks (Facebook, Youtube, Vimeo, Google Calendar) they may not be able to find the right values, keys, and features needed to do so. For example, finding how to set the privacy settings of videos on Vimeo via API: *"How to change a Vimeo's video privacy via API (PHP)? [...] I've followed every step specified by the Vimeo's API Documentation but I can't get it to work. What am I doing wrong?"* [52080930 - 2018]. Another user is looking for which privacy setting are available through the API: *"Which Facebook Privacy settings can be accessed through API? I'm about to start an ASP.NET project which uses Facebook API to get/set Facebook Privacy settings [...] or is there any other way to access other privacy settings, too?"* [9093704 - 2012].

Errors messages can happen when doing things like: accessing restricted resources on an OS, setting the status of posts on social networks, or defining custom access control. The user below is trying to develop a messaging app for iOS with private and public lists, but received an error: *""error:Error Domain=com.quickblox.chat Code=503 "Service not available." So if all privacy list works perfectly then how can my blocked users could send me messages?"* [27665795 - 2016].

I'd like to do it, but how? (31, 10%)

Other entities such as OS or personal drivers lead SO users to ask questions about how to handle access control or provide it to users. For example, *"We develop a rails-based healthcare application. What is the best way to configure our s3 implementation so that only the authenticated user has access to the image?"* [30602560 - 2018].

SO users also look for practices to design databases which provide levels of access control to users: *"Database Design - Users and their privacy [...] It's a good choose? I'm not quite sure if i should create a new table to handle the privacy settings. I must admit that database design isn't my specialty so really need some feedback about this."* [5211799 - 2011]. Askers in this code tend to express personal or "right thing" motivations for adding access control to their databases.

UI elements (18, 6%)

When a privacy dialogue pops up, developers want to get notified about the user's decision so they can react by making changes to the interface or logic of the program: *"It would be a simple thing to reload the view content once the user grants permission, but I'm having a surprisingly hard time finding a method that is called when that happens."* [29338752 - 2015]. Drivers for these questions come from platforms forcing access controls and permission requests.

Theme	*No. Questions*	*Total views*	*Sub-themes (separated by ";")*
Access control	119 (40%)	103,654	Dealing with privacy settings; I'd like to do it, but how?; UI elements; Browsers.
Privacy policies	39 (13%)	127,225	How to do it?; I got an error while trying to implement it; I have got an error in usage descriptions; Do I need a privacy policy? Why?
Encryption	10 (3%)	11,100	How do I achieve it?; Tell me more about it.
Privacy and code issues	5 (2%)	2,523	-
Versions and updates	11 (4%)	22,269	Device and OS versions cause unexpected results; Updates cause unexpected results.
Developers with privacy concerns	71 (24%)	57,136	How to implement privacy?; Can I trust this service or company?; Tell me about it.
Developers ignoring PbD principles	18 (6%)	9,681	-
Developers as end-users	21 (7%)	89,279	How do I protect my data?; Privacy in version control systems (Git); I have privacy concerns, thoughts?

Table 4. Number of questions, total views, and sub-themes for each theme for the 294 qualitatively analysed questions.

Browsers (11, 4%)
Browsers have several features, such as cookie blocking and certificate checking, that are intended to protect users' privacy and security. However, these features can cause issues for developers, such as getting certificate errors when they are using "localhost" during testing, managing cookies, and generating certificates. *"My question is how to setup valid SSL certificate on localhost? or do I need to edit my configuration?"* [35565278 - 2016]. Similarly with cookie blocking: *"If we set on IE11 privacy settings to medium, we succesfully get our value from session, but if we set to "Block All Cookies" - we get null. What can cause it? How to avoid?"* [24059471 - 2014]. The driver for these questions generally comes from browser behaviour and errors.

Privacy policies (39, 13%, views: 127,225)
Developers are often compelled by law, forced by platforms or are personally motivated to provide privacy policies (PPs) to users. SO users ask conceptual questions around PPs as well as more specific questions about how to write them.

How to do it? (15, 5%)
When writing a PP for their apps, SO users have to deal with multiple aspects of composition: wording, technical changes to make their code compliant, effects of third-parties such as analytics libraries, platforms' PP interfaces, and reusing PPs on multiple platforms. For example, complying with GDPR: *"Due to GDPR I am requiring to check the users location whether the user is from European Union"* [50253418 - 2018]. Another user reacted to a news article about Apple's policy against analytics tools and is concerned that their app might be rejected by Apple because of third-party libraries: *"I've just integrated Crashlytics into my code, app is still waiting for review [...] My question is should we be worried by using Crashlytics & Firebase's screen tracking (analytics). Will Apple object it?"* [54658427 - 2019].

I got an error while trying to implement it. (10, 3%)
Questions in this theme deal with errors users got from the platform while trying to publish their app: *"i can't publish my facebook application, when i click "yes" on Status and Reviews of developers platform i see this message "You must provide a valid Privacy Policy URL in order take your app Live. Go to App Details and make sure it is valid." in privacy field i have a right url and i tried also to change it with others but continue to see the messsage. this happens not just for one application but also for others."* [26944634 - 2018].

I have an error in the usage description. (10, 3%)
Apple, in particular, forces "usage description" for accessing restricted resources such as contacts and location. SO users ask questions about errors and crashes they get during development because they do not know how to set these values. They are also confused about messages they receive from Apple after submitting apps without the correct usage descriptions: *"iOS 10 GM release error when submitting apps "app attempts to access privacy-sensitive data without a usage description" due to GoogleSignIn, AdMob."* [39383289 - 2018].

Do I need a privacy policy? Why? (4, 1%)
SO users are confused about why or if a PP is necessary. For simple apps, it can be unclear if a PP is even necessary, or even what the definition of "sensitive data" is. *"My app's operates on a simple couple of button clicks. However, as I am gearing up to release it, I couldn't help but notice nearly all the apps have at least a privacy policy and terms/conditions on there page. Is it legally necessary to have both? Or is it just good practice?"* [56606092 - 2019].

Encryption (10, 3%, views: 11,100)
A fairly small set of questions fall into the encryption theme. Most have a personal motivation or a client requirement.

How do I achieve it? (7, 2%)
Users asked questions about how to implement encryption solutions. *"What could be the best solution to store this data encrypted in a remote database and that only the data's owner could decrypt it? How to make this process transparent to the user? (You can't use the user's password as the key to encrypt his data, because you shouldn't know his password)."* [39772 - 2013]. Questions about encryption errors are also asked: *"I'm using the GnuPG class from PHP. I'm not having any problem importing valid public key but if I try to import something random like "test" which obviously isn't a public key, I'm getting error 502 bad gateway."* [34557651 - 2016].

Tell me more about it. (3, 1%)
These questions ask for further information about encryption solutions. *"Since the salt is used to add a huge range of password possibilities [...] what is the purpose of letting the salt insecure? [...] Is there something that I dont understand? I know that knowing the salt dont break the security but, saying that it "need not be kept secret" sounds strange to me."* [6176848 - 2011].

Privacy and code issues (5, 2%, views: 2,523)
This theme includes errors that are specifically code level and raised due to a function call, security flag, and static analysis tools: *"We use HPE to check the code potential risks, i got one critical issue below in Log util class "The method d() in LogUtil.java mishandles confidential information, which can compromise user privacy and is often illegal". how can i do to fix this?"* [44410004 - 2017].

Versions and updates (11, 4%, views: 22,269)
SO users ask questions when they observe OS and platform behaviours that violate their expectations or desires.

Device and OS versions cause unexpected results. (6, 2%)
Multiple versions for OS and devices can cause frustration for SO users. They test their code on one OS or device, and expect the same behaviour on others. But, this is not always a valid assumption: *"But sometimes iPhone 5s running iOS 8.4 and always iPhone 6 Plus running iOS 9 does not show my app under the privacy photos list."* [32646366 - 2015].

Updates cause unexpected results. (5, 2%)
Updates to OS, platforms, and PPs can be a pain point for SO users: *"I want to give the users of my App the option to control which lists their actions show to by default. The new*

API seems to have taken a feature away because I can't see where that control is!" [7523282 - 2012].

Developers with privacy concerns (71, 24%, views:57,136)
This theme includes questions which are generally in alignment with PbD principles such as minimise, hide, abstract, control, enforce, and inform [35]. Askers in this theme looked for solutions to collect less data, mask personal information, remove unnecessary data, minimise tracking, and other approaches to protect privacy.

These users ask questions about protecting resources such as cookies, location, handwritten documents, browsing habits, IP address, data to build charts and graphs, messages, email address, contacts, Apple ID, phone number, card number, names, health data, country, phone calls, patient health information, personal documents, Facebook activity, images, driver licenses, device IDs, browser history, birth dates, social security numbers, passwords, videos, and the phone's screen.

How to implement privacy? (33, 11%)
Questions in this theme ask about developing privacy-preserving solutions. The motivations come from either personal concerns or requirement from clients. *"My add displays private data, so I don't want it to be possible to see the app contents in the task switcher."* [13260462 - 2012] Similarly: *"I want to mask PII (personal Identification Information) like Name. Birth Date, SSN, Credit card Number, Phone Number, etc. It should remain same formate , means it looks like real data. And shouldn't be reversible. And it should take less time to mask."* [22387577 - 2016].

Can I trust this service or company? (10, 3%)
Specific questions around trusting services are gathered in this theme. The motivations for these questions are either personal or business reasons. For example, when users want to decide to use services (an API or a product) in their projects, they have questions about how much they can trust it with their data and intellectual property: *"Can I trust react-devtools not to breach my privacy? [...] The tool (and react) is made by Facebook, a company infamously known for their complete lack of moral when it comes to data gathering and creepy surveilance of us all. And it requires the ability to access everything you are browsing (which is probably needed to work it's magic), in order to be installed."* [54549807 - 2019].

Tell me about it. (28, 9%)
Conceptual questions around minimising lifetime of data, privacy implications of services (e.g. Google visualisation tools, Google Drive, tracking and cookies, anonymisation). *"Linking to Google PlusOne, without embedding the button (for privacy reasons) It seems that Google only offers code to embed the +1 button. However, there are heavy privacy concerns (plus quite some load time) associated with it."* [9248204 - 2013].

Developers ignoring PbD principles (18, 6%, views: 9,681)
SO users ask questions about workarounds to gain access to data protected by permissions or platform protections. They also have fundamental questions about the reasons for implementing privacy-preserving solutions. They look for access to resources such as: data belonging to other apps, WiFi, Bluetooth, device settings, unique device ID, scores in games, internet and camera permissions, make/model/serial number of computers, screenshots and videos, locations, IP addresses, names, and email address.

SO users ask the community about whether there is a need to do a task with privacy in mind or they can do it without needing privacy permissions. *"How should an app communicate with a server operated by its developer without android.permission.INTERNET? Or is there a reliable source stating that this is impossible in Android?"* [29545251 - 2015].

Some questions looked for instructions on how to collect data, access restricted resources without following proper steps, store sensitive data, combine data from multiple sources, enable cookies, bypass permissions, and identify users. *"How can I read the users computer make, model and serial number from inside MS Edge browser? Using Microsoft Edge web browser, under windows 10, how can I access the make/model and serial number of the computer that the browser is running on?"* [43492726 - 2017].

Developers as end-users (21, 7%, views: 89,279)
Users also ask questions about how to protect the privacy of their own data, software, and identity.

How do I protect my data? (11, 4%)
This theme includes questions around implementing a solution or finding a better approach to protect their own data or intellectual property. *"Whenever I start the program a little eye icon appears in the upper right corner above the scroll bar. It can't be clicked. I assume it's Google uploading my usage data. How can I disable that?"* [19327361 - 2013].

Privacy in version control systems (Git) (8, 3%)
SO users want to protect their source code and identity in version control systems. They also look for suggestions about how to provide access control to projects in these systems. *"Is it possible to completely remove an issue from the GitHub issue tracker?"* [3081521 - 2019] *"What files/folders should I ignore in a git repository of an iOS app? [...] Do the files generated by cocoapods contain some of my private information? Does info.plist file contain my private stuff as well? Also, when I was putting Firebase into my app, I downloaded a GoogleService-Info.plist. Should I ignore it as well? What things should I ignore?"* [37479924 - 2016].

I have privacy concerns, thoughts? (2, 1%)
Questions around personal privacy concerns are grouped here. For personal reasons, SO users look for suggestions to protect their own data in the workspace or from other software companies. *"I recently purchased an advanced chat script which includes free installation on my server. I don't know how to install it but the company says they provide installation if I provide them with the following information: [list of resources to provide access] I don't feel comfortable giving all that info out to them but I know it's required for them to integrate the script to work with my online forum."* [4973811 - 2011].

DISCUSSION
We are not the first to explore how developers think about and interact with privacy concepts. In particular, Hadar et

al. conducted a set of interviews with developers with the aim of understanding their thinking and attitudes around privacy [33]. Similar to our findings, they find that developers often conflate the word "privacy" with security concepts. For example, equating permissions with privacy even though they are technically an access control topic, and have applications beyond privacy. From our own work, we see the conflation of privacy with security potentially coming from the phrasing on platform websites, such as calling permissions "privacy permissions". Developers then learn to equate permissions with the term "privacy". Also, when speaking of privacy, SO users employ language similar to the language of developers in other contexts: encryption, access control, data collection, data removal, data lifetime, and anonymisation are all recurring themes both in our data and in findings from Hadal et al.'s interviews with developers [33].

Of our random question sample (SO-privacy-rand), 17% were conceptual, indicating that developers are looking for advice around privacy-related tasks in the early stages of software development. Such decision-making questions can impact the privacy as well as the security of software: "Security defines which privacy choices can be implemented" [15, p. 669].

Supporting privacy policy creation tasks

While there is research on making PPs understandable for end users [34, 58, 72], there is minimal research on helping developers craft PPs. The lack of support can be seen in the wild, where there are still numerous apps without PPs [77] as well PPs that contain misleading and contradictory statements [7]. In our data, many questions ask for help creating privacy policies. Based on our observations, we hypothesise that some of the problems observed in the wild might be coming from developers who: 1) do not know that they need a PP, 2) do not see a reason for adding a PP, 3) do not know what language needs to be in a PP for their app's unique profile, 4) are trying to add a PP but cannot do it because of complicated procedures as well as unhelpful user interfaces, and 5) see PPs as a wall that is blocking their app being published, with the resulting frustration leading to reluctance to prepare a well written PP.

Developers are sometimes confused about why a PP is needed because they honestly believe that they are not collecting any sensitive data. The developer's understanding of "sensitive" sometimes differed from the platform's definition. Advertising and tracking libraries were another common cause of confusion. Developers were not using sensitive permissions directly, but had included an advertising library which was using some. When they tried publishing their app on an app store, they got a warning that a PP was needed due to sensitive permission usage. They then turned to SO to understand the cause of the issue. Similarly, third-party APIs have privacy implications that need to be reflected in the developer's own PP, with some users turning to SO to figure out exactly what they needed to add to their PP because they used, for example, Firebase which is an app-building infrastructure. The above scenarios exhibit three important themes: 1) the role of platforms in defining what "sensitive" is, 2) the awareness developers have around the types of data their apps collect, and 3) the implications of third-party code and services on PPs.

Writing a PP is a challenging task for developers, especially if they are freelancers or part of small companies with limited legal resources. There is much potential for providing more support to them in this space, particularly automated support which can identify third-party libraries and services in their code and walk them through setting up a PP that correctly describes how data will be collected, stored, and used. This support is particularly needed when the privacy implications of using a service are not immediately obvious. For example, uploading images to Google's image search to find similar images may cause Google to retain and index the uploaded image as Google puts in its PP: "When you upload, submit, store, send or receive content to or through our Services, you give Google (and those we work with) a worldwide license to use, host, store, reproduce..." [28]. Another advantage of automated support is the capability to automatically detect and adapt to changes that occur when third-party PPs change, such as library version updates.

Another possible solution is to better integrate privacy checking into the code development process so that developers can address issues early instead of being rejected when publishing their app or receiving legal complaints after it has been published. Both situations frustrate developers, who feel that they are "done" only to find that they have not yet fulfilled legal obligation. For example, one common cause of app rejection on the Apple Store is "Requesting Permission" without suitable disclosure to the user about permission usage [8].

There are some tools to help with early identification of potential privacy issues due to permission usage. For example, Coconut is an IDE plugin that warns developers during coding when they are dealing with privacy-related tasks such as dealing with user location [41], allowing developers to make any necessary changes earlier in the development process. Such tools could be improved by supporting changes that occur between versions of third-party code. Otherwise, if Apple decides that a new permission is needed to access a specific resource, the developer might only discover the change through experimentation or user complaints.

Platforms as privacy drivers

Platforms such as Apple and Android exert powerful influence on privacy ecosystem [31]. They define the meaning of sensitive content, data, and resource such as camera, contacts, and location [9, 29]. One of the main reasons to ask questions about privacy on SO is rooted in platform requirements. SO users see platforms as gatekeepers for publishing their apps, and perhaps their income source. This gatekeeper role gives platforms the power to enforce privacy behaviour in the applications they host. While some percentage will always try and circumvent, we found that the majority of SO developers were honestly trying to follow the requirements set by platforms.

Platforms also operate as an intermediary between the developer and user on privacy issues. For example, iOS decides when to ask the user about a permission usage and also controls the design of the permission UI the user sees. On one hand, this intermediary role removes a great deal of responsibility from the developer and gives users a more consistent experience. On the other hand, developers loose the ability to

control the full experience of their apps. They also have hand-off related problems when the user is taken to a privacy-related platform screen and then somehow must seamlessly return to the app, even if they have just denied vital permissions.

Shadow documentation

While platforms often provide guidance and documentation around their APIs, libraries, and services, developers still need more specific or targeted guidance. One role SO fills is to provide this documentation through community sourcing answers to specific questions. It is effectively producing a shadow version of official documentation, in a form similar to a Q&A. Parnin et al. studied the Android API documentation and found that 87% of class documentation is also covered on SO [51], making SO a near-complete replacement for consulting official Android documentation. We see similar behaviour with privacy posts: the answers not only include official documentation, but also provide documentation-like information that does not appear elsewhere. Examples include guidance around how to write a PP or how to interpret permissions in relation to existing company guidelines. In effect, SO also hosts community-generated developer-friendly shadow documentation of company policies, PPs, laws, and regulations. On SO, legal-jargon heavy "documentation" is translated into case-specific guidance phrased in a developer-friendly way.

Topic modelling

Topic modelling, which formed the core of our automated analysis, proved an effective way to analyse the entirety of our corpus without the expense or time investment of human annotation. It confirmed impressions of categories from the qualitative annotations, while also pointing to more granular categories and related patterns around language use.

Our manual qualitative analysis and our LDA produced similar high-level results. For example, privacy policies, permission settings, browser errors, and privacy in code repositories come up in both methods. While the obvious difference is scale and time required, there were less-obvious interesting differences such as LDA's natural focus on company names (i.e. Google, Facebook) where the manual coding abstracted these to "platforms". Overall, LDA found topics that are relevant and interesting, with more granularity than the qualitative topics we identified. However, many details of interest were not evident from LDA topics. LDA is a bag-of-words model, meaning that it lacks syntax and semantics in its resulting topics. Consequently the model cannot differentiate issues like if the asker was trying to preserve privacy or intentionally circumvent protections to collect protected data.

Though most of our focus was on qualitative findings, LDA suggested potential avenues of future research. We observed differences across topics in the use of nontechnical vocabulary, like "want", which suggested that some topics are goal-oriented while others are more curiosity-driven and abstract. We also observed differences in the use of polite words like "thanks", which relate to linguistic register, or formality. Differences in register highlight how different topics draw users of disparate technical backgrounds or who project different personas through their language use.

LIMITATIONS

Not all SO users are native English speakers; therefore, we may have misinterpreted some questions because of language issues. Furthermore, we collected questions from SO's full history, hence, some questions may be outdated, though we generally found that while technology aged, the high-level problems remained relevant. SO askers are only occasionally explicit about their driver for posting. Drivers such as compiler errors, or platform requirements are clear from text, but motivation-style drivers like personal or client are very challenging to differentiate cleanly. While the difference would be interesting, we cannot provide it with high confidence.

When starting our qualitative analysis process, we reviewed three privacy frameworks [35, 62, 64] to create a group grounding for the term "privacy". While we ultimately decided to use SO users' own definition of the word, this early review may have impacted our analysis.

FUTURE WORK

Prior work has shown that traces of SO code snippets are visible in the apps that people use. Potential future work might look at traces of SO's answers in app PPs. A further step in understanding privacy on SO is to explore answers and questions together to understand the dynamics between users and how they build knowledge around privacy-related topics. Developers in small companies who integrate ad networks for monetisation view advertisement companies as being responsible for user privacy [46]; our work points to similar questions about how developers view app stores and themselves in relation to users' privacy. Experiments with LDA point to distinct nontechnical language use in different topics; future work could look at politeness, formality, and other aspects of persona associated with different privacy topics, possibly investigating what questions are asked by different communities or skill levels of programmers.

CONCLUSION

We analysed privacy-related questions on SO with LDA and qualitative analysis. Our results show that SO users face challenges while writing and modifying privacy policies; working with or designing systems with access control; dealing with updates to platforms and APIs; and deciding on privacy aspects of their projects. Platforms can use these results to improve the privacy-related workflows to create an experience that is efficient and convenient. Google, Apple, and Facebook are privacy influencers who define what content is considered sensitive, and are major drivers that bring developers to SO to ask privacy questions. Any of these entities have the ability to impact how developers think about and interact with privacy and impact the privacy ecosystem of software.

ACKNOWLEDGEMENTS

We thank everyone associated with the TULiPS Lab at the University of Edinburgh for helpful discussions and feedback. We also thank the anonymous reviewers whose comments helped improve the paper greatly. This work was sponsored in part by Microsoft Research through its PhD Scholarship Program and a Google Research Award.

REFERENCES

[1] Yasemin Acar, Michael Backes, Sascha Fahl, Simson Garfinkel, Doowon Kim, Michelle L Mazurek, and Christian Stransky. 2017. Comparing the Usability of Cryptographic APIs. In *2017 IEEE Symposium on Security and Privacy (SP)*. 154–171. DOI: http://dx.doi.org/10.1109/SP.2017.52

[2] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. 2016. You Get Where You're Looking for: The Impact of Information Sources on Code Security. In *2016 IEEE Symposium on Security and Privacy (SP)*. 289–305. DOI:http://dx.doi.org/10.1109/SP.2016.25

[3] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. 2017. How Internet Resources Might Be Helping You Develop Faster but Less Securely. *IEEE Security Privacy* 15, 2 (March 2017), 50–60. DOI: http://dx.doi.org/10.1109/MSP.2017.24

[4] Alexa. 2019. stackoverflow.com. Retrieved September 2019 from https://www.alexa.com/siteinfo/stackoverflow.com

[5] Miltiadis Allamanis and Charles Sutton. 2013. Why, when, and what: Analyzing Stack Overflow questions by topic, type, and code. In *2013 10th Working Conference on Mining Software Repositories (MSR)*. 53–56. DOI: http://dx.doi.org/10.1109/MSR.2013.6624004

[6] Le An, Ons Mlouki, Foutse Khomh, and Giuliano Antoniol. 2017. Stack Overflow: A code laundering platform?. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. 283–293. DOI: http://dx.doi.org/10.1109/SANER.2017.7884629

[7] Benjamin Andow, Samin Yaseer Mahmud, Wenyu Wang, Justin Whitaker, William Enck, Bradley Reaves, Kapil Singh, and Tao Xie. 2019. PolicyLint: Investigating Internal Privacy Policy Contradictions on Google Play. In *28th USENIX Security Symposium (USENIX Security 19)*. USENIX Association, Santa Clara, CA, 585–602. https://www.usenix.org/conference/usenixsecurity19/presentation/andow

[8] Apple App Store. 2019. App Review. Retrieved September 2019 from https://developer.apple.com/app-store/review

[9] Apple Developer Documentation. 2019. Accessing Protected Resources. Retrieved September 2019 from https://developer.apple.com/documentation/uikit/protecting_the_user_s_privacy/accessing_protected_resources

[10] Hala Assal and Sonia Chiasson. 2018. Security in the Software Development Lifecycle. In *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. USENIX Association, Baltimore, MD, 281–296. https://www.usenix.org/conference/soups2018/presentation/assal

[11] Hala Assal and Sonia Chiasson. 2019. 'Think Secure from the Beginning': A Survey with Software Developers. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 289, 13 pages. DOI: http://dx.doi.org/10.1145/3290605.3300519

[12] Hala Assal, Sonia Chiasson, and Robert Biddle. 2016. Cesar: Visual representation of source code vulnerabilities. In *2016 IEEE Symposium on Visualization for Cyber Security (VizSec)*. 1–8. DOI: http://dx.doi.org/10.1109/VIZSEC.2016.7739576

[13] Rebecca Balebako and Lorrie Cranor. 2014. Improving App Privacy: Nudging App Developers to Protect User Privacy. *IEEE Security Privacy* 12, 4 (July 2014), 55–58. DOI:http://dx.doi.org/10.1109/MSP.2014.70

[14] Rebecca Balebako, Abigail Marsh, Jialiu Lin, Jason I Hong, and Lorrie Cranor. 2014. The privacy and security behaviors of smartphone app developers. In *Workshop on Usable Security (USEC'14)*. Internet Society. DOI: http://dx.doi.org/10.14722/usec.2014.23006

[15] Derek E. Bambauer. 2013. Privacy versus Security. *Journal of Criminal Law and Criminology* 103 (2013), 667–684. https://ssrn.com/abstract=2208824

[16] Anton Barua, Stephen W. Thomas, and Ahmed E. Hassan. 2014. What are developers talking about? An analysis of topics and trends in Stack Overflow. *Empirical Software Engineering* 19, 3 (Jun 2014), 619–654. DOI: http://dx.doi.org/10.1007/s10664-012-9231-y

[17] Kathrin Bednar, Sarah Spiekermann, and Marc Langheinrich. 2019. Engineering Privacy by Design: Are engineers ready to live up to the challenge? *The Information Society* 35, 3 (2019), 122–142. DOI: http://dx.doi.org/10.1080/01972243.2019.1583296

[18] Stefanie Beyer and Martin Pinzger. 2014. A Manual Categorization of Android App Development Issues on Stack Overflow. In *2014 IEEE International Conference on Software Maintenance and Evolution*. 531–535. DOI: http://dx.doi.org/10.1109/ICSME.2014.88

[19] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3 (March 2003), 993–1022. http://dl.acm.org/citation.cfm?id=944919.944937

[20] Ann Cavoukian. 2009. Privacy by design: The 7 foundational principles. *Information and Privacy Commissioner of Ontario, Canada* 5 (2009). https://iab.org/wp-content/IAB-uploads/2011/03/fred_carter.pdf

[21] Ann Cavoukian, Scott Taylor, and Martin E. Abrams. 2010. Privacy by Design: essential for organizational accountability and strong business practices. *Identity in the Information Society* 3, 2 (01 Aug 2010), 405–413. DOI:http://dx.doi.org/10.1007/s12394-010-0053-z

[22] Lisa Nguyen Quang Do, Karim Ali, Benjamin Livshits, Eric Bodden, Justin Smith, and Emerson Murphy-Hill. 2017. Just-in-time Static Analysis. In *Proceedings of the 26th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA 2017)*. ACM, New York, NY, USA, 307–317. DOI: http://dx.doi.org/10.1145/3092703.3092705

[23] Paul Dourish and Ken Anderson. 2006. Collective Information Practice: Exploring Privacy and Security as Social and Cultural Phenomena. *Human-Computer Interaction* 21, 3 (2006), 319–342. DOI: http://dx.doi.org/10.1207/s15327051hci2103_2

[24] Manuel Egele, David Brumley, Yanick Fratantonio, and Christopher Kruegel. 2013. An Empirical Study of Cryptographic Misuse in Android Applications. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS '13)*. ACM, New York, NY, USA, 73–84. DOI: http://dx.doi.org/10.1145/2508859.2516693

[25] Sascha Fahl, Marian Harbach, Henning Perl, Markus Koetter, and Matthew Smith. 2013. Rethinking SSL Development in an Appified World. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS '13)*. ACM, New York, NY, USA, 49–60. DOI: http://dx.doi.org/10.1145/2508859.2516655

[26] Felix Fischer, Konstantin Böttinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. 2017. Stack Overflow Considered Harmful? The Impact of Copy Paste on Android Application Security. In *2017 IEEE Symposium on Security and Privacy (SP)*. 121–136. DOI: http://dx.doi.org/10.1109/SP.2017.31

[27] Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov. 2012. The Most Dangerous Code in the World: Validating SSL Certificates in Non-browser Software. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS '12)*. ACM, New York, NY, USA, 38–49. DOI: http://dx.doi.org/10.1145/2382196.2382204

[28] Google. 2019. Privacy & Terms. Retrieved September 2019 from https://policies.google.com/terms

[29] Google Developers. 2019. Manifest.permission. Retrieved September 2019 from https://developer.android.com/reference/android/Manifest.permission.html

[30] Matthew Green and Matthew Smith. 2016. Developers Are Not the Enemy!: The Need for Usable Security APIs. *IEEE Security and Privacy* 14, 5 (Sept. 2016), 40–46. DOI:http://dx.doi.org/10.1109/MSP.2016.111

[31] Daniel Greene and Katie Shilton. 2018. Platform privacies: Governance, collaboration, and the different meanings of "privacy" in iOS and Android development. *New Media & Society* 20, 4 (2018), 1640–1657. DOI: http://dx.doi.org/10.1177/1461444817702397

[32] Seda Gürses, Carmela Troncoso, and Claudia Diaz. 2011. Engineering privacy by design. *Computers, Privacy & Data Protection* 14, 3 (2011), 25. https://software.imdea.org/~carmela.troncoso/papers/Gurses-CPDP11.pdf

[33] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. 2018. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering* 23, 1 (Feb 2018), 259–289. DOI: http://dx.doi.org/10.1007/s10664-017-9517-1

[34] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G. Shin, and Karl Aberer. 2018. Polisis: Automated Analysis and Presentation of Privacy Policies Using Deep Learning. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, Baltimore, MD, 531–548. https://www.usenix.org/conference/usenixsecurity18/presentation/harkous

[35] Jaap-Henk Hoepman. 2019. *Privacy Design Strategies (The Little Blue Book)*. Radboud University. https://cs.ru.nl/~jhh/publications/pds-booklet.pdf

[36] Nasif Imtiaz, Akond Rahman, Effat Farhana, and Laurie Williams. 2019. Challenges with Responding to Static Analysis Tool Alerts. In *Proceedings of the 16th International Conference on Mining Software Repositories (MSR '19)*. IEEE Press, Piscataway, NJ, USA, 245–249. DOI: http://dx.doi.org/10.1109/MSR.2019.00049

[37] Shubham Jain, Janne Lindqvist, and others. 2014. Should I protect you? Understanding developers' behavior to privacy-preserving APIs. In *Workshop on Usable Security (USEC'14)*. Internet Society. DOI: http://dx.doi.org/10.14722/usec.2014.23045

[38] Brittany Johnson, Yoonki Song, Emerson Murphy-Hill, and Robert Bowdidge. 2013. Why Don't Software Developers Use Static Analysis Tools to Find Bugs?. In *Proceedings of the 2013 International Conference on Software Engineering (ICSE '13)*. IEEE Press, Piscataway, NJ, USA, 672–681. DOI: http://dx.doi.org/10.1109/ICSE.2013.6606613

[39] Bert-Jaap Koops, Jaap-Henk Hoepman, and Ronald Leenes. 2013. Open-source intelligence and privacy by design. *Computer Law & Security Review* 29, 6 (2013), 676 – 688. http://www.sciencedirect.com/science/article/pii/S0267364913001672

[40] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. 2017. Chapter 8 - Interviews and focus groups. In *Research Methods in Human Computer Interaction* (second edition ed.), Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser (Eds.). Morgan Kaufmann, Boston, 187 – 228. DOI: http://dx.doi.org/10.1016/B978-0-12-805390-4.00008-X

[41] Tianshi Li, Yuvraj Agarwal, and Jason I. Hong. 2018. Coconut: An IDE Plugin for Developing Privacy-Friendly Apps. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 178 (Dec. 2018), 35 pages. DOI: http://dx.doi.org/10.1145/3287056

[42] Yuanchun Li, Fanglin Chen, Toby Jia-Jun Li, Yao Guo, Gang Huang, Matthew Fredrikson, Yuvraj Agarwal, and Jason I. Hong. 2017. PrivacyStreams: Enabling Transparency in Personal Data Processing for Mobile Apps. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 76 (Sept. 2017), 26 pages. DOI: http://dx.doi.org/10.1145/3130941

[43] Tamara Lopez, Thein Tun, Arosha Bandara, Mark Levine, Bashar Nuseibeh, and Helen Sharp. 2019. An Anatomy of Security Conversations in Stack Overflow. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS '10)*. IEEE Press, Piscataway, NJ, USA, 31–40. DOI: http://dx.doi.org/10.1109/ICSE-SEIS.2019.00012

[44] Tamara Lopez, Thein T. Tun, Arosha Bandara, Mark Levine, Bashar Nuseibeh, and Helen Sharp. 2018. An Investigation of Security Conversations in Stack Overflow: Perceptions of Security and Community Involvement. In *Proceedings of the 1st International Workshop on Security Awareness from Design to Deployment (SEAD '18)*. ACM, New York, NY, USA, 26–32. DOI:http://dx.doi.org/10.1145/3194707.3194713

[45] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. 2011. Design Lessons from the Fastest Q&A Site in the West. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2857–2866. DOI: http://dx.doi.org/10.1145/1978942.1979366

[46] Abraham H. Mhaidli, Yixin Zou, and Florian Schaub. 2019. "We Can't Live Without Them!" App Developers' Adoption of Ad Networks and Their Considerations of Consumer Risks. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. USENIX Association, Santa Clara, CA. https://www.usenix.org/conference/soups2019/presentation/mhaidli

[47] Sarah Nadi, Stefan Krüger, Mira Mezini, and Eric Bodden. 2016. Jumping Through Hoops: Why Do Java Developers Struggle with Cryptography APIs?. In *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*. 935–946. DOI: http://dx.doi.org/10.1145/2884781.2884790

[48] Seyed Mehdi Nasehi, Jonathan Sillito, Frank Maurer, and Chris Burns. 2012. What makes a good code example?: A study of programming Q A in StackOverflow. In *2012 28th IEEE International Conference on Software Maintenance (ICSM)*. 25–34. DOI:http://dx.doi.org/10.1109/ICSM.2012.6405249

[49] Duc Cuong Nguyen, Dominik Wermke, Yasemin Acar, Michael Backes, Charles Weir, and Sascha Fahl. 2017. A Stitch in Time: Supporting Android Developers in WritingSecure Code. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17)*. ACM, New York, NY, USA, 1065–1077. DOI: http://dx.doi.org/10.1145/3133956.3133977

[50] OWASP. 2017. *Top 10 - 2017 The ten most critical web application security risks*. Technical Report. The OWASP Foundation. https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

[51] Chris Parnin, Christoph Treude, Lars Grammel, and Margaret-Anne Storey. 2012. Crowd documentation: Exploring the coverage and the dynamics of API discussions on Stack Overflow. *Georgia Institute of Technology, Tech. Rep* 11 (2012). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.371.6263

[52] Nikhil Patnaik, Joseph Hallett, and Awais Rashid. 2019. Usability Smells: An Analysis of Developers' Struggle With Crypto Libraries. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS)*. USENIX Association. https://www.usenix.org/conference/soups2019/presentation/patnaik

[53] Olgierd Pieczul, Simon Foley, and Mary Ellen Zurko. 2017. Developer-centered Security and the Symmetry of Ignorance. In *Proceedings of the 2017 New Security Paradigms Workshop (NSPW 2017)*. ACM, New York, NY, USA, 46–56. DOI: http://dx.doi.org/10.1145/3171533.3171539

[54] Chaiyong Ragkhitwetsagul, Jens Krinke, Matheus Paixao, Giuseppe Bianco, and Rocco Oliveto. 2019. Toxic Code Snippets on Stack Overflow. *IEEE Transactions on Software Engineering* (2019), 1–22. DOI:http://dx.doi.org/10.1109/TSE.2019.2900307

[55] Akond Rahman, Asif Partho, Patrick Morrison, and Laurie Williams. 2018. What Questions Do Programmers Ask About Configuration As Code?. In *Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering (RCoSE '18)*. ACM, New York, NY, USA, 16–22. DOI: http://dx.doi.org/10.1145/3194760.3194769

[56] Christoffer Rosen and Emad Shihab. 2016. What are mobile developers asking about? A large scale study using stack overflow. *Empirical Software Engineering* 21, 3 (Jun 2016), 1192–1223. DOI: http://dx.doi.org/10.1007/s10664-015-9379-3

[57] Neil Salkind. 2019. Encyclopedia of Research Design. (Sept. 2019). DOI: http://dx.doi.org/10.4135/9781412961288

[58] Florian Schaub, Rebecca Balebako, and Lorrie Faith Cranor. 2017. Designing Effective Privacy Notices and Controls. *IEEE Internet Computing* 21, 3 (May 2017), 70–77. DOI:http://dx.doi.org/10.1109/MIC.2017.75

[59] Awanthika Senarath and Nalin A. G. Arachchilage. 2018. Why Developers Cannot Embed Privacy into Software Systems?: An Empirical Investigation. In *Proceedings of the 22Nd International Conference on Evaluation and Assessment in Software Engineering 2018 (EASE'18)*. ACM, New York, NY, USA, 211–216. DOI:http://dx.doi.org/10.1145/3210459.3210484

[60] Katie Shilton and Daniel Greene. 2019. Linking Platforms, Practices, and Developer Ethics: Levers for Privacy Discourse in Mobile Application Development. *Journal of Business Ethics* 155, 1 (Mar 2019), 131–146. DOI:http://dx.doi.org/10.1007/s10551-017-3504-8

[61] H. Jeff Smith, Tamara Dinev, and Heng Xu. 2011. Information Privacy Research: An Interdisciplinary Review. *MIS Q.* 35, 4 (Dec. 2011), 989–1016. http://dl.acm.org/citation.cfm?id=2208940.2208950

[62] Daniel J Solove. 2005. A taxonomy of privacy. *University of Pennsylvania Law Review* 154 (2005), 477–560. https://ssrn.com/abstract=667622

[63] spaCy. 2019. spaCy - Industrial-strength Natural Language Processing in Python. Retrieved September 2019 from https://spacy.io

[64] Sarah Spiekermann and Lorrie Faith Cranor. 2009. Engineering Privacy. *IEEE Transactions on Software Engineering* 35, 1 (Jan 2009), 67–82. DOI: http://dx.doi.org/10.1109/TSE.2008.88

[65] Stack Exchange. 2019. Stack Exchange Data Explorer. Retrieved September 2019 from https://data.stackexchange.com

[66] Stack Overflow. 2019a. About. Retrieved September 2019 from https://stackoverflow.com/company

[67] Stack Overflow. 2019b. Developer Survey Results. Retrieved August 2019 from https://insights.stackoverflow.com/survey/2019

[68] Stack Overflow. 2019c. What does it mean when an answer is "accepted"? Retrieved September 2019 from https://stackoverflow.com/help/accepted-answer

[69] Stack Overflow. 2019d. Where Developers Learn, Share, & Build Careers. Retrieved September 2019 from https://stackoverflow.com

[70] Mohammad Tahaei and Kami Vaniea. 2019. A Survey on Developer-Centred Security. In *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*. 129–138. DOI: http://dx.doi.org/10.1109/EuroSPW.2019.00021

[71] Christoph Treude, Ohad Barzilay, and Margaret-Anne Storey. 2011. How Do Programmers Ask and Answer Questions on the Web? (NIER Track). In *Proceedings of the 33rd International Conference on Software Engineering (ICSE '11)*. ACM, New York, NY, USA, 804–807. DOI: http://dx.doi.org/10.1145/1985793.1985907

[72] Yung Shin Van Der Sype and Walid Maalej. 2014. On lawful disclosure of personal user data: What should app developers do?. In *2014 IEEE 7th International Workshop on Requirements Engineering and Law (RELAW)*. 25–34. DOI: http://dx.doi.org/10.1109/RELAW.2014.6893479

[73] Richmond Y. Wong and Deirdre K. Mulligan. 2019. Bringing Design to the Privacy Table: Broadening "Design" in "Privacy by Design" Through the Lens of HCI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 262, 17 pages. DOI: http://dx.doi.org/10.1145/3290605.3300492

[74] Yuhao Wu, Shaowei Wang, Cor-Paul Bezemer, and Katsuro Inoue. 2019. How do developers utilize source code from stack overflow? *Empirical Software Engineering* 24, 2 (Apr 2019), 637–673. DOI: http://dx.doi.org/10.1007/s10664-018-9634-5

[75] Jing Xie, Heather Richter Lipford, and Bill Chu. 2011. Why do programmers make security errors?. In *2011 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 161–164. DOI: http://dx.doi.org/10.1109/VLHCC.2011.6070393

[76] Xin-Li Yang, David Lo, Xin Xia, Zhi-Yuan Wan, and Jian-Ling Sun. 2016. What Security Questions Do Developers Ask? A Large-Scale Study of Stack Overflow Posts. *Journal of Computer Science and Technology* 31, 5 (Sep 2016), 910–924. DOI: http://dx.doi.org/10.1007/s11390-016-1672-0

[77] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel Reidenberg, N. Cameron Russell, and Norman Sadeh. 2019. MAPS: Scaling Privacy Compliance Analysis to a Million Apps. *Proceedings on Privacy Enhancing Technologies* 2019, 3 (2019), 66 – 86. DOI: http://dx.doi.org/10.2478/popets-2019-0037